Martch 2006
UWThPh-2006-8
hep-th/0603220

Graphical Representation of SUSY and C-Program Calculation

Shoichi ICHINOSE *

Institut für Theoretische Physik, Universität Wien

Boltzmanngasse 5, A-1090 Vienna, Austria

Abstract

We present a graphical representation of the supersymmetry and a C-program for the graphical calculation. Calculation is demonstrated for 4D Wess-Zumino model and for Super QED. The chiral operators are graphically expressed in an illuminating way. The tedious part of SUSY calculation, due to manipulating chiral suffixes, reduces considerably. The application is diverse.

PACS NO: 02.10.Ox, 02.70.-c, 02.90.+p, 11.30.Pb, 11.30.Rd

Key Words: Graphical representation, Supersymmetry, Spinor suffix, Chiral suffix, Graph index, suffix contraction, C-program

1 Introduction

The supersymmetry is the symmetry between fermions and bosons. It was introduced in the mid 70's. At present the experiment does not yet confirm the symmetry, but everybody accepts its importance in nature and expects fruitful results in the future developement. The requirement of such a high symmetry costs a sophisticated structure which makes its dynamical analysis difficult. In this circumstance, we propose a calculational technique which utilizes the graphical representation of SUSY. The representation was proposed in [1, 2]. [1] The spinor is represented as a slanted line with a direction.

*On leave of absence from Laboratory of Physics, School of Food and Nutritional Sciences, University of Shizuoka. Untill 31 March, 2006.
E-mail address: ichinose@u-shizuoka-ken.ac.jp

[1] An improved version of Ref.[1] has recently appeared as Ref.[3].

Its chirality is represented by the way the line is drawn. The introductory explantion is given in the text. The advantage of the graph expression is the use of the *graph indices*. Every independent graph, which corresponds to a unique term in the ordinary calculation, is classified by a set of graph indices. Hence the main efforts of programinng is devoted to find good graph indices and to count them. SUSY calculation generally is not a simple algebraic or combinatoric or analytical one. It involves the vast branch of mathematics including Grassmann algebra. The delicate property of chirality is produced in this environment. Hence it seems that the ordinary(popular) programs, such as Mathematica, Maple, REDUCE, MAXIMA, form, do not work. It requires a more fundamental language. We take C-language and present a first-step program. Future development to the level of the previously cited ordinary programs is much expected.

The notation in the text is based on the standard textbook by Wess and Bagger[4].

2 Spinors in SUSY: Graphical Representation and Storage Form

Weyl spinors have the $\mathrm{SU}(2)_L \times \mathrm{SU}(2)_R$ structure. The *chiral* suffix α, appearing in ψ^α or ψ_α, represents (fundamental representation, doublet representation) $\mathrm{SU}(2)_L$ and the *anti-chiral* suffix $\dot{\alpha}$, appearing in $\bar{\psi}^{\dot{\alpha}}$ or $\bar{\psi}_{\dot{\alpha}}$, represents $\mathrm{SU}(2)_R$. The raising and lowering of suffixes are done by the antisymmetric tensors $\epsilon^{\alpha\beta}$ and $\epsilon_{\alpha\beta}$.

$$(\epsilon^{\alpha\beta}) = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad , \quad (\epsilon_{\dot{\alpha}\dot{\beta}}) = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \quad , \quad \epsilon^{\alpha\beta}\epsilon_{\beta\gamma} = \delta^\alpha_\gamma \quad ,$$
$$\psi^\alpha = \epsilon^{\alpha\beta}\psi_\beta \quad , \quad \bar{\psi}_{\dot{\alpha}} = \epsilon_{\dot{\alpha}\dot{\beta}}\bar{\psi}^{\dot{\beta}} \quad . \qquad (1)$$

They are graphically expressed by Fig.1. We encode them as follows. We use 2 dimensional array with the size 2×2. The four chiral spinors are stored in C-program as the array psi[ ][ ].

Fermionic Fields

(1) (Weyl) Spinor [Symbol: p ; Dimension: $\mathrm{M}^{3/2}$]

ψ^α	ψ_α	$\bar{\psi}_{\dot{\alpha}}$	$\bar{\psi}^{\dot{\alpha}}$
psi[0,0]=α	psi[0,0]=empty	psi[0,0]=empty	psi[0,0]=empty
psi[0,1]=empty	psi[0,1]=α	psi[0,1]=empty	psi[0,1]=empty
psi[1,0]=empty	psi[1,0]=empty	psi[1,0]=empty	psi[1,0]=$\dot{\alpha}$
psi[1,1]=empty	psi[1,1]=empty	psi[1,1]=$\dot{\alpha}$	psi[1,1]=empty

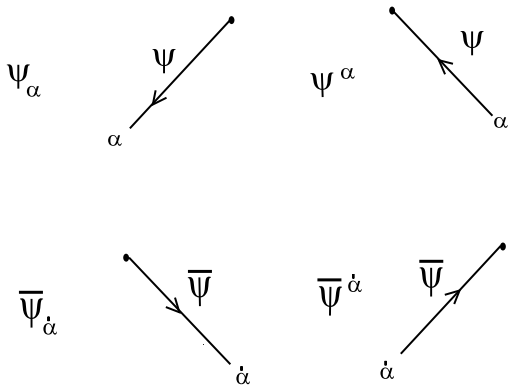


Figure 1: Weyl fermions.

The first column takes two numbers 0 and 1; 0 expresses a 'chiral' operator ψ, while 1 expresses an 'anti-chiral' operator $\bar{\psi}$. The second column also takes the two numbers; 0 expresses an 'up' suffix, while 1 expresses an 'down' one.

Note: Chiral spinor suffixes $\alpha, \beta, \cdots$ are expressed , in the present C-program, by positive *odd* number integers 1,3,$\cdots$, while anti-chiral ones $\dot{\alpha}, \dot{\beta}, \cdots$ are by positive *even* number integers 2,4,$\cdots$. This convension (discriminative use of even and odd integers) is, at this stage, rather redundant in the sense that the chirality (ψ or $\bar{\psi}$) can be read by the first element number of the array psi[2][2] for a non-empty data, i.e. 0 for ψ (psi[0][*]=α) and 1 for $\bar{\psi}$ (psi[1][*]=$\dot{\alpha}$). (The situation is the same for some other spinors $\partial_m\psi, \theta, \cdots$. See the later discription.) The convension, however, will soon become important to discriminate the chirality of the spinor matrices; σ and $\bar{\sigma}$.

Note: 'empty' is expressed by a default number (, for example, 99) in the program.

(2) First derivative of spinor [Symbol: q ; Dimension: $M^{5/2}$]
The first derivative of the spinor is graphically expressed by the upper graph of Fig.2. It is stored by one 2×2 array dps[ ][ ] and one variable dpsv.

$\partial_m\psi^\alpha$	$\partial_m\psi_\alpha$	$\partial_m\bar{\psi}_{\dot{\alpha}}$	$\partial_m\bar{\psi}^{\dot{\alpha}}$
dps[0,0]=α	dps[0,0]=empty	dps[0,0]=empty	dps[0,0]=empty
dps[0,1]=empty	dps[0,1]=α	dps[0,1]=empty	dps[0,1]=empty
dps[1,0]=empty	dps[1,0]=empty	dps[1,0]=empty	dps[1,0]=$\dot{\alpha}$
dps[1,1]=empty	dps[1,1]=empty	dps[1,1]=$\dot{\alpha}$	dps[1,1]=empty
dpsv=m	dpsv=m	dpsv=m	dpsv=m

Here the vector suffix expresses the Lorents suffix of a differetial operator.

Note: The vector suffixes m, n, $\cdots$ are expressed by 51, 52, $\cdots$ in the

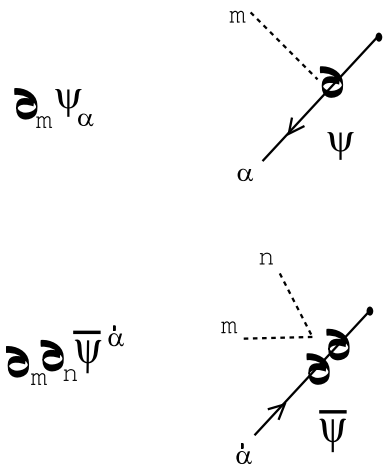


Figure 2: Derivatives of fermions.

present program.

(3) Sigma Matrix [Symbol: s ; Dimension: M^0]
Sigma matrices $\sigma^m, \bar{\sigma}^m$ are graphically expressed in Fig.3.
They are stored as the 2×2 array si[ ][ ].

$\sigma^m_{\alpha\dot{\alpha}}$	$\bar{\sigma}^{m\dot{\alpha}\alpha}$
si[0,0]=empty	si[0,0]=$\dot{\alpha}$
si[0,1]=α	si[0,1]=empty
si[1,0]=empty	si[1,0]=α
si[1,1]=$\dot{\alpha}$	si[1,1]=empty
siv=m	siv=m

Note: The use of even (α) and odd ($\dot{\alpha}$) integers makes an important role here. The arrangement of spinor suffixes, that is (left-side suffix, right-side suffix)=(odd, even) or (even, odd), makes us clear the difference between σ and $\bar{\sigma}$. We should, however, have the relation

$$\bar{\sigma}^m_{\dot{\alpha}\alpha} = \sigma^m_{\alpha\dot{\alpha}}$$

in mind. Hence the above 2quantities are equivelently expressed as

$\bar{\sigma}^m_{\dot{\alpha}\alpha}$	$\sigma^{m\alpha\dot{\alpha}}$
si[0,0]=empty	si[0,0]=α
si[0,1]=$\dot{\alpha}$	si[0,1]=empty
si[1,0]=empty	si[1,0]=$\dot{\alpha}$
si[1,1]=α	si[1,1]=empty
siv=m	siv=m

This ambiguity does not cause any problem because we keep a rule:

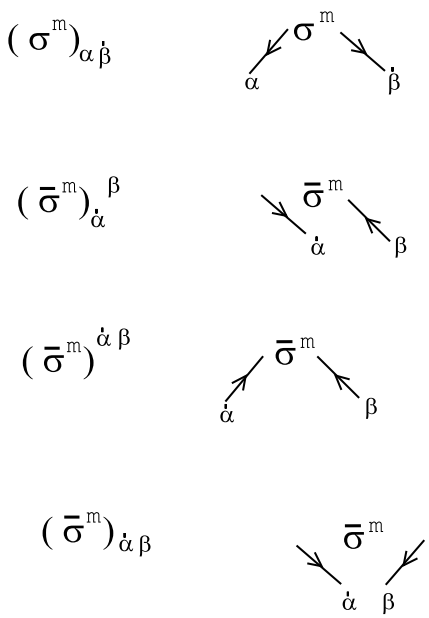


Figure 3: Elements of SL(2,C) σ-matrices. $(\sigma^m)_{\alpha\dot{\beta}}$ and $(\bar{\sigma}^m)^{\dot{\alpha}\beta}$ are the standard form.

- In this program, we use only σ (not use $\bar{\sigma}^m$).

- $\bar{\sigma}^m$ is used only for the graphical explanation.

(4) Superspace coordinate [Symbol: t ; Dimension: $M^{-1/2}$]
The superspace coordinate θ^α is exprssed in the same way as the spinor ψ^α.

θ^α	θ_α	$\bar{\theta}_{\dot{\alpha}}$	$\bar{\theta}^{\dot{\alpha}}$
th[0,0]=α	th[0,0]=empty	th[0,0]=empty	th[0,0]=empty
th[0,1]=empty	th[0,1]=α	th[0,1]=empty	th[0,1]=empty
th[1,0]=empty	th[1,0]=empty	th[1,0]=empty	th[1,0]=$\dot{\alpha}$
th[1,1]=empty	th[1,1]=empty	th[1,1]=$\dot{\alpha}$	th[1,1]=empty

They are graphically expressed by Fig.4.

(5) Gagino [Symbol: l ; Dimension: $M^{3/2}$]
The photino λ^α is exprssed in the same way as the spinor ψ^α. We take the 2×2 array la[ ][ ].

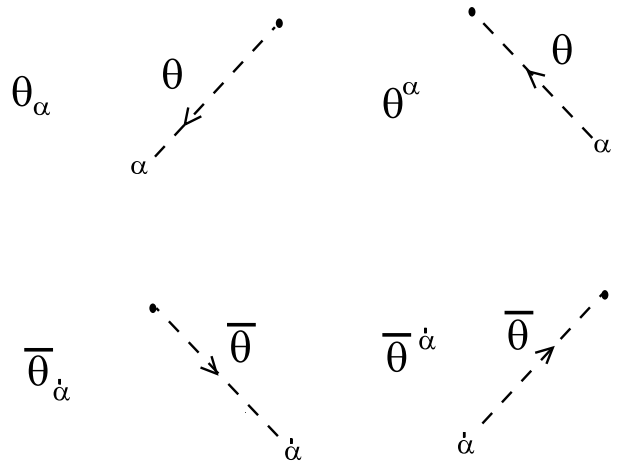


Figure 4: The graphical representation for the spinor coordinates in the superspace: $\theta_\alpha, \theta^\alpha, \bar{\theta}_{\dot{\alpha}}$ and $\bar{\theta}^{\dot{\alpha}}$.

λ^α	λ_α	$\bar{\lambda}_{\dot{\alpha}}$	$\bar{\lambda}^{\dot{\alpha}}$
la[0,0]=α	la[0,0]=empty	la[0,0]=empty	la[0,0]=empty
la[0,1]=empty	la[0,1]=α	la[0,1]=empty	la[0,1]=empty
la[1,0]=empty	la[1,0]=empty	la[1,0]=empty	la[1,0]=$\dot{\alpha}$
la[1,1]=empty	la[1,1]=empty	la[1,1]=$\dot{\alpha}$	la[1,1]=empty

(6) The first derivative of gaugino [Symbol: m ; Dimension: $M^{5/2}$]
The first derivative of the photino is expressed as the 2×2 array dl[ ][ ] and the variable dlv.

$\partial_m\lambda^\alpha$	$\partial_m\lambda_\alpha$	$\partial_m\bar{\lambda}_{\dot{\alpha}}$	$\partial_m\bar{\lambda}^{\dot{\alpha}}$
dl[0,0]=α	dl[0,0]=empty	dl[0,0]=empty	dl[0,0]=empty
dl[0,1]=empty	dl[0,1]=α	dl[0,1]=empty	dl[0,1]=empty
dl[1,0]=empty	dl[1,0]=empty	dl[1,0]=empty	dl[1,0]=$\dot{\alpha}$
dl[1,1]=empty	dl[1,1]=empty	dl[1,1]=$\dot{\alpha}$	dl[1,1]=empty
dlv=m	dlv=m	dlv=m	dlv=m

Bosonic Fields

(7) Complex scalar [Symbol: A ; Dimension: M^1]
The complex scalar field A is expressed by one dimensional array A[ ] with 2 elements.

A	A^*
A[0]=1(exist)	A[0]=empty
A[1]=empty	A[1]=1(exist)

where the element-numbers 0,1 correspond to A(chiral) or A^*(anti-chiral), respectively.

(8) The first derivative of the complex scalar [Symbol: B ; Dimension: M^2]
The first derivative of A and A^* are expressed as the one dimensional array B[ ] with 2 elements.

∂_mA	∂_mA*
B[0]=m	B[0]=empty
B[1]=empty	B[1]=m

(9) Vector field [Symbol: v ; Dimension: M^1]
The vector field(photon) v^m is expressed by one variable v.

v^m
v=m
v=empty(non-exist)
The lowest expression is taken when the vector field does not appear.

(10) The first derivative of the vector field [Symbol: w ; Dimension: M^2]
The first derivative of v^m is expressed by two variables dv and dvv.

$\partial_n v_m$
dv=m
dvv=n

(11) The Dalemberian derivative of A and A* [Symbol: C ; Dimension: M^3]
The Dalemberian derivative of A and A^* are expressed as the one dimensional array C[ ] with 2 elements.

$\partial_m\partial^m$A	$\partial_m\partial^m$A*
C[0]=1(exist)	C[0]=empty
C[1]=empty	C[1]=1(exist)

(12) Auxiliary fields [Symbol: F ; Dimension: M^2]
The auxiliary fields F and F^* are expressed as

F	F*
F[0]=1(exist)	F[0]=empty
F[1]=empty	F[1]=1(exist)

(13) Real auxilary fields [Symbol: D ; Dimension: M^2]
The auxiliary field D (real scalar), which appears in the vector multiplet, is expressed as
D
D=1(exist)
D=empty(non-exist)

Term and Component

Terms in the SUSY calculation are stored as an ordered set of the above quantities. The following examples appear in the intermidiate stage of evaluating $W_\alpha W^\alpha$ where W_α is the field strength superfield. (See App.B.)

Example 1

The term $\sigma^l_{\delta\dot\alpha}\sigma^{m\beta\dot\alpha}\sigma^{s\delta}{}_{\dot\gamma}\sigma^n{}_\beta{}^{\dot\gamma}\partial_l v_m\partial_s v_n = -\sigma^l_{\delta\dot\alpha}\bar\sigma^{m\dot\alpha\beta}\sigma^n_{\beta\dot\gamma}\bar\sigma^{s\dot\gamma\delta}\partial_l v_m\partial_s v_n$ is stored, in the computer, as follows.

type[c=0]=s	type[c=1]=s	type[c=2]=s	type[c=3]=s	
si[c=0,0,1]=1	si[c=1,0,0]=3	si[c=2,0,0]=1	si[c=3,0,1]=3	
si[c=0,1,1]=2	si[c=1,1,0]=2	si[c=2,1,1]=4	si[c=3,1,0]=4	continued
siv[c=0]=51	siv[c=1]=52	siv[c=2]=53	siv[c=3]=54	below

type[c=4]=w	type[c=5]=w
dv[c=4]=52	dv[c=5]=54
dvv[c=4]=51	dvv[c=5]=53

($\delta\to 1, \dot\alpha\to 2, \beta\to 3, \dot\gamma\to 4;\ l\to 51, m\to 52, s\to 53, n\to 54$)

Besides the above ones, the number of components (compno=6) and an overall *weight* (complex) are necessary to characterize a term. One additional coloumn, specified by the variable c, appears. The variable c controls the order of every *component*. This *ordering* is important in the calculation involving *Grassmannian* quantities. In the above storing form, each component starts with specifying the type: s, w, $\cdots$. The one dimensional array type[ ] is used for the purpose. This term graphically appears in the output (before further reduction) as shown in Fig.5.

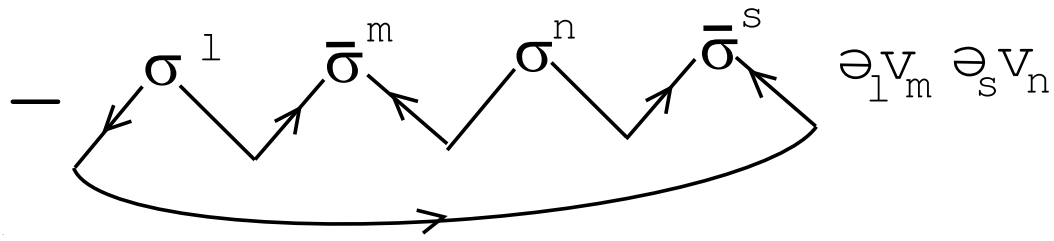


Figure 5: The graphical expression of $-\sigma^l_{\delta\dot\alpha}\bar\sigma^{m\dot\alpha\beta}\sigma^n_{\beta\dot\gamma}\bar\sigma^{s\dot\gamma\delta}\partial_l v_m\partial_s v_n$.

In Fig.5, we see all dummy suffixes disappear and the advantage of the graphical expression is manifest. The chirality can be read from the shape of the directed-line graph.

Example 2

The term $\sigma^{l\delta}_{\dot\alpha}\sigma^m_\delta{}^{\dot\alpha} D\partial_l v_m = -\sigma^l_{\delta\dot\alpha}\bar\sigma^{m\dot\alpha\delta} D\partial_l v_m$ is stored, in the computer, as follows.

type[c=0]=s	type[c=1]=s		
si[c=0,0,0]=1	si[c=1,0,1]=1		type[c=3]=w
si[c=0,1,1]=4	si[c=1,1,0]=4	type[c=2]=D	dv[c=3]=54
siv[c=0]=53	siv[c=1]=54	D[c=2]=1	dvv[c=3]=53

($\delta \to 1, \dot{\alpha} \to 4;\ l \to 53, m \to 54$)
In this case, the number of components is 4 (compno=4) and there is a weight. This one graphically appears in the output (before further reduction) as shown in Fig.6.

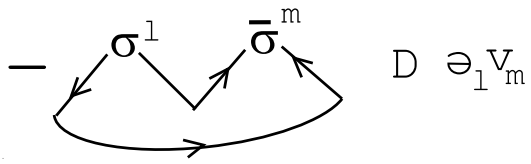


Figure 6: The graphical expression of $-\sigma^l_{\delta\dot{\alpha}}\bar{\sigma}^{m\dot{\alpha}\delta}D\partial_l v_m$.

These two examples clearly show the advantage of the graph representation over the conventional one using indicies. The new representation discriminates each term not by the suffixes but by the shape of the graph. It means the importance of indices of the graph such as the number of the chiral-loop. (In the above examples, the number is 1.)

3 Field name, id-number and Grassmannian property

One advantage of C-language is the efficient manipulation of characters. The control of characters is important for specifying various operators (fields) appearing in SUSY theories. As described previously, we assign every operator a symbol which is expressed by a character. The appearance of many fields, which is a big cause complicating SUSY theories, can be neatly handled by the use of the character name. We also introduce the id-number for every operator. (See the list below)

For every operator, we assign the Grassmann number: +1(commuting) or -1(anti-commuting). This assignment is exploited in the process of *moving a component* , within a term, *respecting the Grassmannian property of operators*.

We list the assignment in TABLE 1 with the dimension of operators.

Fid[0]='t';	DIM[0]=0.0;	Grassmann[0]=-1;
Fid[1]='s';	DIM[1]=0.0;	Grassmann[1]=1;
/* <Chiral */		
Fid[100]='p';	DIM[100]=(float)3/2;	Grassmann[100]=-1;
Fid[101]='q';	DIM[101]=(float)5/2;	Grassmann[101]=-1;
Fid[102]='A';	DIM[102]=1.0;	Grassmann[102]=1;
Fid[103]='B';	DIM[103]=2.0;	Grassmann[103]=1;
Fid[104]='C';	DIM[104]=3.0;	Grassmann[104]=1;
Fid[105]='F';	DIM[105]=2.0;	Grassmann[105]=1;
/* <Chiral2 */		
Fid[120]='P';	DIM[120]=(float)3/2;	Grassmann[120]=-1;
Fid[121]='Q';	DIM[121]=(float)5/2;	Grassmann[121]=-1;
Fid[122]='a';	DIM[122]=1.0;	Grassmann[122]=1;
Fid[123]='b';	DIM[123]=2.0;	Grassmann[123]=1;
Fid[124]='c';	DIM[124]=3.0;	Grassmann[124]=1;
Fid[125]='f';	DIM[125]=2.0;	Grassmann[125]=1;
/* <Vector */		
Fid[140]='l';	DIM[140]=(float)3/2;	Grassmann[140]=-1;
Fid[141]='m';	DIM[141]=(float)5/2;	Grassmann[141]=-1;
Fid[142]='D';	DIM[142]=2.0;	Grassmann[142]=1;
Fid[143]='v';	DIM[143]=1.0;	Grassmann[143]=1;
Fid[144]='w';	DIM[144]=2.0;	Grassmann[144]=1;

TABLE 1 Definition of FieldID, Dimension, Grassmann No

4 Graph Indices: vpairno, NcpairO, NcpairE, closed-chiral-loop-No, GrNum

SigmaContraction (B),(C),(D) and (E)

In the process of SUSY calculation, there appear graphs connected by directed lines (chiral suffixes contraction) and by (non-directed) dotted lines (vector suffixes contraction). We can classify them by some *graph indices*.

vpairno The number of vector-suffix contractions.

NcpairO The number of chiral-suffix contractions. This is equal to the number of left-directed wedges.

NcpairE The number of anti-chiral-suffix contractions. This is equal to the number of the right-directed wedges.

closed-chiral-loop-No The closed-chiral-loop is the case that the directed lines , connected by σ or $\bar{\sigma}$, make a loop. In this case NcpairO=NcpairE. The number of closed chiral loops is defined to this index.

GrNum A group is defined to be a set of σ's or $\bar{\sigma}$'s which are connected by directed lines. The number of groups is defined to be GrNum.

In TABLE 2-4, we list the classification of the product of σ's using the graph indices defined above.

These tables clearly show the σ-matrices play an important role to connect the chiral world and the space-time (Lorentz) world.

5 Superspace coordinates

Supersymmetry is most manifestly expressed in the superspace $(x^m, \theta, \bar{\theta})$. $\theta_\alpha = \epsilon_{\alpha\beta}\theta^\beta$, $\bar{\theta}^{\dot{\alpha}} = \epsilon^{\dot{\alpha}\dot{\beta}}\bar{\theta}_{\dot{\beta}}$ are spinorial coordinates. They satisfy the relations graphically shown in Fig.7. These relations are exploited in the program in order to sort the SUSY quantities with respect to the power of $\theta\theta$ and $\bar{\theta}\bar{\theta}$. (For further detail, see the subsection thBthBthth of App.A.)

6 Treatment of Metrics: $\epsilon_{\alpha\beta}$, $\epsilon_\alpha^{\ \beta} = \epsilon^\beta_{\ \alpha} = \delta^\beta_\alpha$, η^{mn} and the totally anti-symmetric tensor ϵ^{lmns}

For the totally anti-symmetric tensor ϵ^{lmns}, we introduce one dimensional array ep[ ] with 4 components.

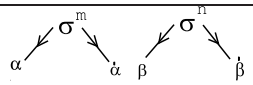
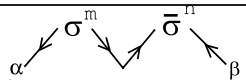
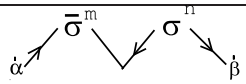
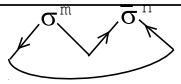
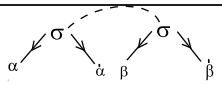
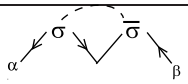
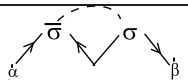
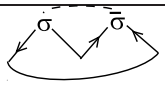
vpairno	NcpairO	NcpairE	figure
0	0	0	
	0	1	
	1	0	
	1	1	 $= -2\eta^{mn}$
1	0	0	 $= -2\epsilon_{\alpha\beta}\epsilon_{\dot{\alpha}\dot{\beta}}$
	0	1	 $= -4\delta^{\beta}_{\alpha}$
	1	0	 $= -4\delta^{\dot{\alpha}}_{\dot{\beta}}$
	1	1	 $= -8$

TABLE 2 Classification of the product of 2 sigma matrices (nsi=2).

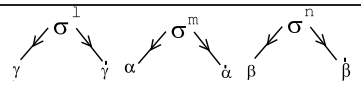
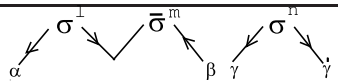
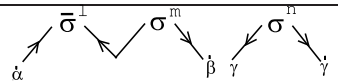
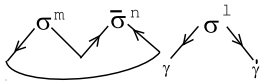
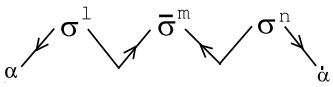
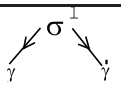
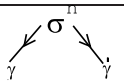
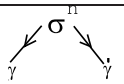
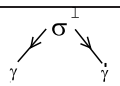
vpairno	NcpairO	NcpairE	figure
0	0	0	
	0	1	
	1	0	
	1	1	closed-chiral-loop No =1 
			closed-chiral-loop No =0 
1	0	0	$-2\epsilon_{\alpha\beta}\epsilon_{\dot\alpha\dot\beta}$ 
	0	1	$-4\delta^{\beta}_{\alpha}$ 
	1	0	$-4\delta^{\dot\alpha}_{\dot\beta}$ 
	1	1	-8 

TABLE 3 Classification of the product of 3 sigma matrices (nsi=3).

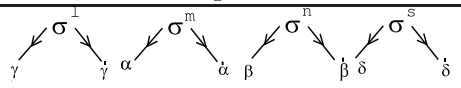
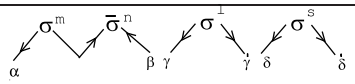
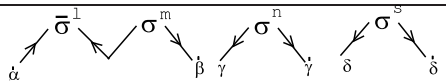
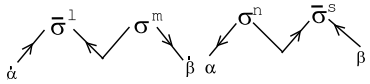
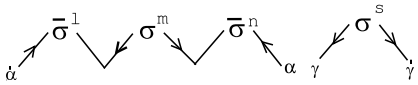
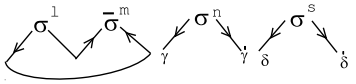
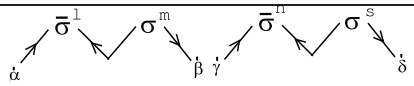
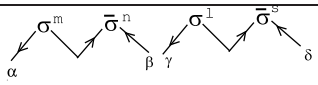
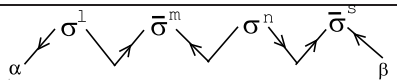
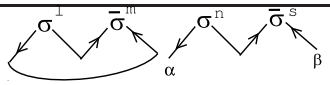
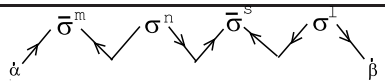
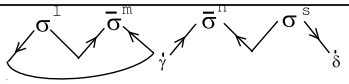
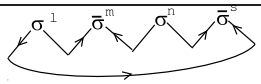
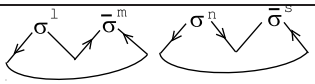
vpairno	NcpairO	NcpairE	figure
0	0	0	
	0	1	
	1	0	
	1	1	GrNum=2, Division=(2,2) 
			GrNum=2, Division=(3,1) 
			GrNum=3, Division=(2,1,1) 
	2	0	
	0	2	
	1	2	GrNum=1, 
			GrNum=2, 
	2	1	GrNum=1, 
			GrNum=2, 
	2	2	GrNum=1, 
			GrNum=2, 

TABLE 4 Classification of the product of 4 sigma matrices with no vector-suffix contraction (nsi=4, vpairno=0).

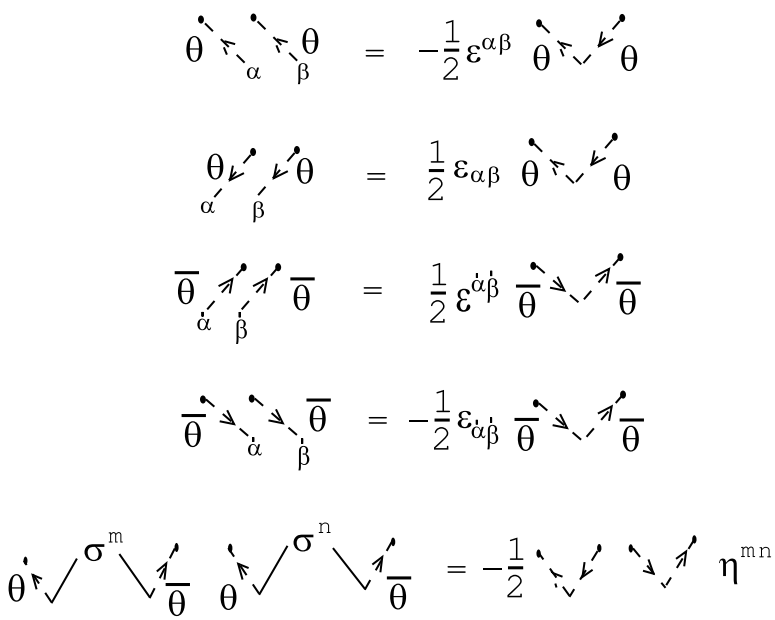


Figure 7: The graphical rules for the spinor coordinates: $\theta^\alpha\theta^\beta = -\frac{1}{2}\epsilon^{\alpha\beta}\theta\theta$, $\theta_\alpha\theta_\beta = \frac{1}{2}\epsilon_{\alpha\beta}\theta\theta$, $\bar\theta^{\dot\alpha}\bar\theta^{\dot\beta} = \frac{1}{2}\epsilon^{\dot\alpha\dot\beta}\bar\theta\bar\theta$, $\bar\theta_{\dot\alpha}\bar\theta_{\dot\beta} = -\frac{1}{2}\epsilon_{\dot\alpha\dot\beta}\bar\theta\bar\theta$, $\theta\sigma^m\bar\theta\theta\sigma^n\bar\theta = -\frac{1}{2}\theta\theta\bar\theta\bar\theta\eta^{mn}$.

ep[0]=l
ep[1]=m
ep[2]=n
ep[3]=s

Symbol: e

This term appears in Sec.7 and produces topologically important terms such as $v_{lm}\tilde v^{lm} = \epsilon^{lmns}v_{lm}v_{ns}$.

As for the metric of the chiral suffix, we do *not* introduce specific arrays. They play a role of raising or lowering suffixes, which can be encoded in the upper (0) and lower (1) code in arrays. For the Lorentz metric η^{mn}, we do not need to much care for the discrimination between the upper and lower suffixes because of the even-symmetry with respect to the change of the Lorentz suffixes($\eta^{mn} = \eta^{nm}$).

7 Sigma matrices

Let us express important relations valid between products of sigma matrices graphically. In Fig.8, the symmetric combination of $\bar\sigma^m\sigma^n$ are shown as the basic spinor algebra. The antisymmetric combination gives the generators of

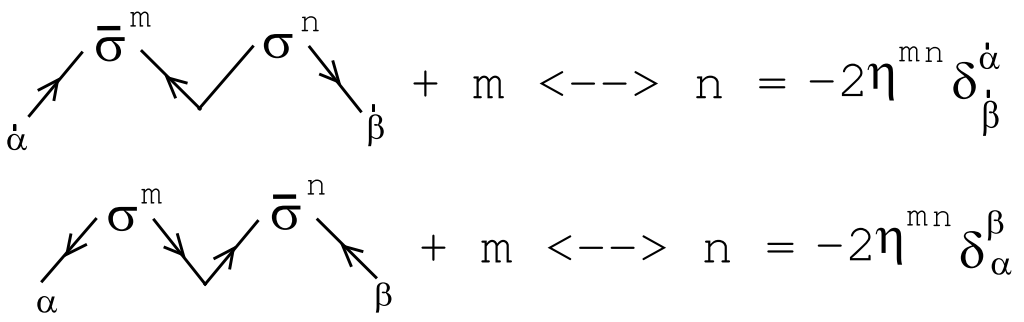


Figure 8: A graphical formula of the basic spinor algebra. $(\bar{\sigma}^m)^{\dot{\alpha}\beta}(\sigma^n)_{\beta\dot{\beta}} + m \leftrightarrow n = -2\eta^{mn}\delta^{\dot{\alpha}}_{\dot{\beta}}$, and $(\sigma^m)_{\alpha\dot{\alpha}}(\bar{\sigma}^n)^{\dot{\alpha}\beta} + m \leftrightarrow n = -2\eta^{mn}\delta^{\beta}_{\alpha}$.

the Lorentz group, $\sigma^{nm}, \bar{\sigma}^{nm}$.

$$(\sigma^{nm})_\alpha^{\ \beta} = \frac{1}{4}\left\{ \quad - m \leftrightarrow n \right\} \quad ,$$

$$(\bar{\sigma}^{nm})^{\dot{\alpha}}_{\ \dot{\beta}} = \frac{1}{4}\left\{ \quad - m \leftrightarrow n \right\} \quad , \qquad (2)$$

The "reduction" formulae (from the cubic σ's to the linear one) are expressed as in Fig.9. From Fig.9, we notice any chain of σ's can always be expressed by less than three σ's. The appearance of the 4th rank anti-symmetric tensor ϵ^{lmns} is quite illuminating. The *completeness* relations are expressed as in Fig.10. The contraction, expressed by arrowed curve in Fig.10.is the matrix trace.

Fierz identity is graphically shown as follows.

$$= \frac{1}{4}\left\{ - \quad \epsilon_{\dot{\alpha}\dot{\beta}} + \quad \epsilon_{\alpha\beta} - m \leftrightarrow n \right\}$$

$$-\frac{1}{2}\eta^{nm}\epsilon_{\alpha\beta}\epsilon_{\dot{\alpha}\dot{\beta}} - \frac{1}{8}\left\{ \quad - l \leftrightarrow n \right\}\left\{ \quad - l \leftrightarrow m \right\}, \qquad (3)$$

These relations are important, in the program, to reduce the product of sigma matrices. Especially at the place (E) of SigmaContraction, they are exploited. As an example, the closed chiral-loop graph in Fig.5 reduces as follows.

$$= 2(\eta^{lm}\eta^{ns} - \eta^{ln}\eta^{ms} + \eta^{ls}\eta^{mn} - i\epsilon^{lmns}) \qquad (4)$$

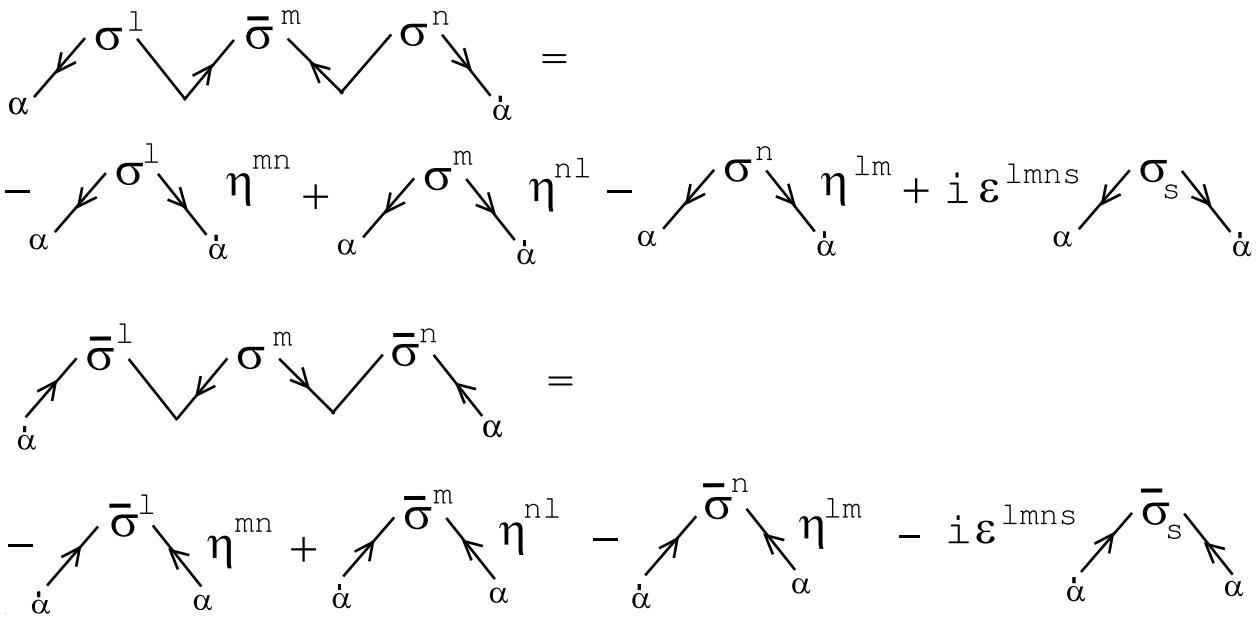


Figure 9: Two relations: 1) $\sigma^l\bar{\sigma}^m\sigma^n = -\sigma^l\eta^{mn} + \sigma^m\eta^{nl} - \sigma^n\eta^{lm} + i\epsilon^{lmns}\sigma_s$, 2) $\bar{\sigma}^l\sigma^m\bar{\sigma}^n = -\bar{\sigma}^l\eta^{mn} + \bar{\sigma}^m\eta^{nl} - \bar{\sigma}^n\eta^{lm} - i\epsilon^{lmns}\bar{\sigma}_s$.

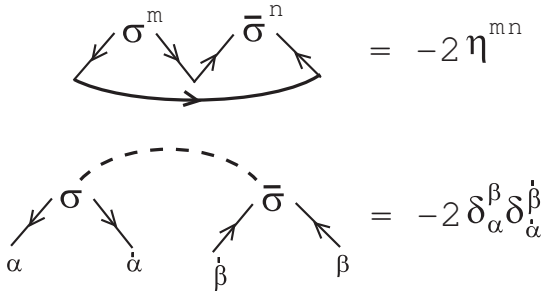
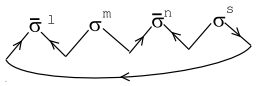


Figure 10: Completeness relations: 1) $\delta^\alpha_\beta(\sigma^m)_{\alpha\dot{\alpha}}(\bar{\sigma}^n)^{\dot{\alpha}\beta} = -2\eta^{mn}$, 2) $(\sigma^m)_{\alpha\dot{\alpha}}(\bar{\sigma}_m)^{\dot{\beta}\beta} = -2\delta^\beta_\alpha\delta^{\dot{\beta}}_{\dot{\alpha}}$. The contraction using δ^α_β is the matrix trace. The relation 1) is also obtained from Fig.8.

Its complex conjugate one is given by

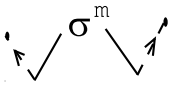


$$= 2(\eta^{lm}\eta^{ns} - \eta^{ln}\eta^{ms} + \eta^{ls}\eta^{mn} + i\epsilon^{lmns}) \qquad (5)$$

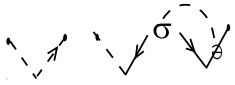
The above 2 terms appear in the calculation of $W_\alpha W^\alpha$ and $\bar{W}_{\dot{\alpha}}\bar{W}^{\dot{\alpha}}$ respectively. (W_α and $\bar{W}_{\dot{\alpha}}$ are superfields of field strength.)

8 Superfield and Structure of Input Data

The transformation between the superfield expression and the fields-components expression is an important subject of SUSY theories. For the purpose, we do the calculation of $\Phi^\dagger\Phi$, in the App.B. In this case, the input data is taken from the content of the superfield. The first superfield $\Phi^\dagger$(we assign the superfield-number sf=0) and the second one Φ(sf=1) have 6 terms (we assign the term-number t=0,1,$\cdots$,5) within each superfield. The input data, written in App.B, can be read from the following expression.

$\Phi^\dagger = -i$ 

$\partial_m A^* +$  $\frac{1}{4}\partial^2 A^* + 2$ 

$+i$ 

$+$  $F^* + A^*$ (6)

This data of $\Phi^\dagger$ is stored as

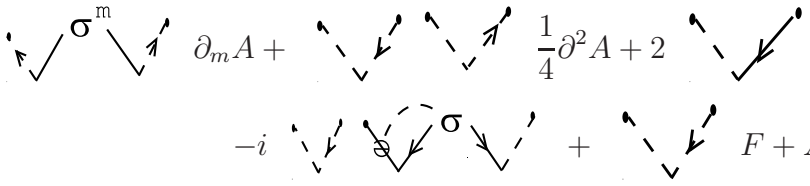
weight[sf=0,t=0]=0+i(-1)	weight[sf=0,t=1]=1+i(0)
type[sf=0,t=0,c=0]= t	type[sf=0,t=1,c=0]= t
th[sf=0,t=0,c=0,0,0]=1	th[sf=0,t=1,c=0,0,0]=1
type[sf=0,t=0,c=1]= s	type[sf=0,t=1,c=1]= t
si[sf=0,t=0,c=1,0,1]=1	th[sf=0,t=1,c=1,0,1]=1
si[sf=0,t=0,c=1,1,1]=2	type[sf=0,t=1,c=2]= t
siv[sf=0,t=0,c=1]=51	th[sf=0,t=1,c=2,1,1]=2
type[sf=0,t=0,c=2]= t	type[sf=0,t=1,c=3]= t
th[sf=0,t=0,c=2,1,0]=2	th[sf=0,t=1,c=3,1,0]=2
type[sf=0,t=0,c=3]= B	type[sf=0,t=1,c=4]= C
B[sf=0,t=0,c=3,1]=51	C[sf=0,t=1,c=4,1]=1 $\cdots$

Here we notice two additional coloumns, specified by sf and t, appear. They specify the order of the supersields and the order of the terms within each superfield respectively.

The superfield Φ is graphically shown as

$$\Phi = i \quad \partial_m A + \quad \frac{1}{4}\partial^2 A + 2$$

$$-i \quad + \quad F + A \qquad (7)$$



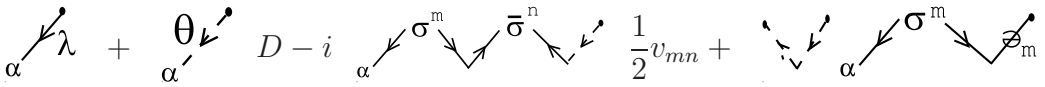
This data of Φ is stored as

weight[sf=1,t=0]=0+i(1)	weight[sf=1,t=1]=1+i(0)
type[sf=1,t=0,c=0]= t	type[sf=1,t=1,c=0]= t
th[sf=1,t=0,c=0,0,0]=5	th[sf=1,t=1,c=0,0,0]=5
type[sf=1,t=0,c=1]= s	type[sf=1,t=1,c=1]= t
si[sf=1,t=0,c=1,0,1]=5	th[sf=1,t=1,c=1,0,1]=5
si[sf=1,t=0,c=1,1,1]=6	type[sf=1,t=1,c=2]= t
siv[sf=1,t=0,c=1]=52	th[sf=1,t=1,c=2,1,1]=6
type[sf=1,t=0,c=2]= t	type[sf=1,t=1,c=3]= t
th[sf=1,t=0,c=2,1,0]=6	th[sf=1,t=1,c=3,1,0]=6
type[sf=1,t=0,c=3]= B	type[sf=1,t=1,c=4]= C
B[sf=1,t=0,c=3,0]=52	C[sf=1,t=1,c=4,0]=1 ⋯

The calculation of $\Phi\dagger\Phi$ leads to the Wess-Zumino Lagrangian. (See App.B)

We also do the calculation of $W_\alpha W^\alpha$ in App.B where W_α is the field strength superfield. They are expressed as follows.

$$W_\alpha = -i \quad + \quad D - i \quad \frac{1}{2}v_{mn} + \quad (8)$$

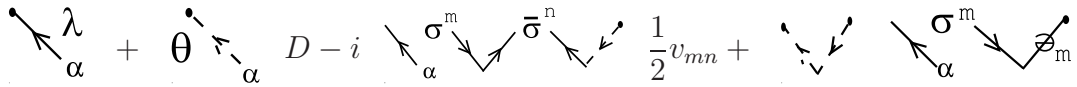


This data of W_α is stored as

	weight[sf=0,t=1]=1+i(0)
	type[sf=0,t=1,c=0]= t
weight[sf=0,t=0]=0+i(-1)	th[sf=0,t=1,c=0,0,1]=1
type[sf=0,t=0,c=0]= l	type[sf=0,t=1,c=1]= D
la[sf=0,t=0,c=0,0,1]=1	D[sf=0,t=1,c=1]=1 ⋯

W^α is expressed as

$$W^\alpha = -i \quad + \quad D - i \quad \frac{1}{2}v_{mn} + \quad (9)$$



This data of W^α is stored as

	weight[sf=1,t=1]=1+i(0)
	type[sf=1,t=1,c=0]= t
weight[sf=1,t=0]=0+i(-1)	th[sf=1,t=1,c=0,0,0]=1
type[sf=1,t=0,c=0]= l	type[sf=1,t=1,c=1]= D
la[sf=1,t=0,c=0,0,0]=1	D[sf=1,t=1,c=1]=1 $\cdots$

The calculation of $W_\alpha W^\alpha$ leads to the Super Electromagnetism Lagrangian.(See App.B)

9 Conclusion

In the history of the quantum field theory, new techniques have produced physically important results. The regularization techniques are such examples. The dimensional regularization by 'tHooft and Veltman[5] produced important results on the renormalization group property of Yang-Mills theory and many scattering amplitude calculations. The lattice regularization in the gauge theory revealed non-perturbative features of hadron physics. In this case, the computor technique of numerical calculation is essential. As for the computer algebraic one, we recall the calculation of 2-loop on-shell counterterms of pure Einstein gravity[6, 7]. A new technique is equally important as a new idea.

The SUSY theory is beautifully constructed respecting the symmetry between bosons and fermions, but the attractiveness is practically much reduced by its complicated structure: many fields, chiral properties, Grassmannian algebra, etc. The present approach intends to improve the situation by a computer program which makes use of the graphical technique. (This approach is taken in Ref.[8] for the calculation of product of SO(N) tensors. It was applied to various anomaly calculations.)

The present program should be much more improved. Here we cite the prospective final goal.

1. It can do the transformation between the superfield expression and the component expression.

2. It can do the SUSY trnasformation of various quantities. In particular it can confirm the SUSY-invariance of the Lagrangian in the graphical way and give the final total divergence.

3. It can do algebraic SUSY calculation involving $D_\alpha, \bar{D}^{\dot{\alpha}}, Q_\alpha$ and $\bar{Q}^{\dot{\alpha}}$.

The item 1 above has been demostrated in the present paper.

It is impossible to deal with all SUSY calculations. This is simply because which fields appear and which dimensional quantities are calculated

depend on each problem. If we obatin a list of (graph) indices which classify all phsical quantities (operators) appearing in the output, then the present program works (by adding new lines for the new problem). To deal with such a case, we add the appendix B where the program flow is explained. It intends to help the reader to read the original source code.

10 Appendix A: Skech of Programming Flow

[2]

[2]For simplicity, we omit the vector multiplet components.

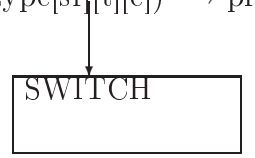
10.1 Input of Data

[Initialization of Basic Components]
sf=0,1,$\cdots$maxsfieldno-1; t=0,1,$\cdots$maxtermno-1; c=0,1,$\cdots$maxcompno-1; i,j,k=0,1
type[sf][t][c]='x'
th[sf][t][c][i][j]=99
psi[sf][t][c][i][j]=99
dps[sf][t][c][i][j]=99
dpsv[sf][t][c]=99
si[sf][t][c][i][j]=99
siv[sf][t][c]=99
A[sf][t][c][k]=99
B[sf][t][c][k]=99
C[sf][t][c][k]=99
F[sf][t][c][k]=99

sfieldno=2
for (sf=0;sf<sfieldno;sf++)
{ /* <sf-running */
printf("SUPER FIELD NO" ,sf)
fscanf (& termno)$\longrightarrow$ printf, noterm[sf]
for (t=0;t<termno;t++)
{ /* <t-running */
printf("term=",t)

[Initialization of Graph Indices]
dif=0, nth=thn=thbarn=0, npsi=psin=psibarn=0, nsi=sin=sibarn=0
nA=An=Abarn=0, nB=Bn=Bbarn=0, nC=Cn=Cbarn=0, nF=Fn=Fbarn=0

fscanf (&weight[sf][t][2])$\longrightarrow$ printf
fscanf (&compno)$\longrightarrow$ printf, nocomp[sf][t]
for (c=0;c<compno;c++)
{ /* <c-running */
printf(c)
fscanf (&type[sf][t][c])$\longrightarrow$ printf



[Install of SuperField Data]

case 't' /* Spinor Coordinate $\theta, \bar{\theta}$ */

nth→ nth+1 printf(nth)

fscanf(&lr,&ud, &spsuf)

switch(lr)→ { lr=0 thn → thn+1 printf(thn)
lr=1 thbarn → thbarn+1 printf(thbarn)

printf (spsuf)

th[sf][t][c][lr][ud]=spsuf

case 'p' /* Spinor $\psi, \bar{\psi}$ */

npsi→ npsi+1 printf(npsi)

fscanf(&lr,&ud, &spsuf)

switch(lr)→ { lr=0 psin → psin+1 printf(psin)
lr=1 psibarn → psibarn+1 printf(psibarn)

printf (spsuf)

psi[sf][t][c][lr][ud]=spsuf

case 'q' /* der Spinor $\partial\psi, \partial\bar{\psi}$ */

npsi→ npsi+1, dif→dif+1 printf(npsi)

fscanf(&lr,&ud, &spsuf)

switch(lr)→ { lr=0 psin → psin+1 printf(psin)
lr=1 psibarn → psibarn+1 printf(psibarn)

printf (spsuf)

dps[sf][t][c][lr][ud]=spsuf

fscan(&vecsuf)

dpsv[sf][t][c]=vecsuf

case 's' /* sigma $\sigma, \bar{\sigma}$ */

nsi→ nsi+1 printf(nsi)

fscanf(&lr1,&ud1, &spsuf1)

fscanf(&lr2,&ud2, &spsuf2)

$\begin{cases} \text{lr1=0,lr2=1 (case'901')} & \text{sin} \rightarrow \text{sin+1, printf} \\ \text{lr1=1,lr2=0 (case'910')} & \text{sibarn} \rightarrow \text{sibarn+1, printf} \end{cases}$

si[sf][t][c][lr1][ud1]=spsuf1

si[sf][t][c][lr2][ud2]=spsuf2

fscanf(&vecsuf)

siv[sf][t][c]=vecsuf

case 'B' /* der Scalar $\partial A, \partial \bar{A}$ */

nB→ nB+1, dif→dif+1 printf(nB)

fscanf(&lr, &vecsuf)

$\begin{cases} \text{lr=0 } (\partial A) & \text{Bn} \rightarrow \text{Bn+1, printf(Bn)} \\ \text{lr=1 } (\partial A^*) & \text{Bbarn} \rightarrow \text{Bban+1, printf(Bbarn)} \end{cases}$

B[sf][t][c][lr]=vecsuf, printf

case 'A' /* Scalar $A, \bar{A}$ */

nA→ nA+1, printf(nA)

fscanf(&lr, &existno)

$\begin{cases} \text{lr=0 } (A) & \text{An} \rightarrow \text{An+1, printf(An)} \\ \text{lr=1 } (A^*) & \text{Abarn} \rightarrow \text{Abarn+1, printf(Abarn)} \end{cases}$

A[sf][t][c][lr]=existno, printf

case 'C' /* der de Scalar $\partial\partial A, \partial\partial \bar{A}$ */

nC→ nC+1, dif→ dif+2, printf(nC)

fscanf(&lr, &existno)

$\begin{cases} \text{lr=0 } (\partial\partial A) & \text{Cn} \rightarrow \text{Cn+1, printf(Cn)} \\ \text{lr=1 } (\partial\partial A^*) & \text{Cbarn} \rightarrow \text{Cbarn+1, printf(Cbarn)} \end{cases}$

C[sf][t][c][lr]=existno, printf

```
    } /* c-running> */
} /* t-running> */
} /* sf-running> */
```

/* Expanding the product of superfields */

T[0]=0,1,$\cdots$,noterm[0]-1: Term No of Superfield 0
T[1]=0,1,$\cdots$,noterm[1]-1: Term No of Superfield 1
.
.
T[sfieldno-1]=0,1,$\cdots$,noterm[sfieldno-1]-1: Term No of Superfield sfieldno-1

For every set (T[0],T[1],$\cdots$,T[sfieldno-1]), do the following box.
printf (T[0],T[1],$\cdots$,T[sfieldno-1])

```
termscombine();
thn=THnum1[0]; thbarn=THnum1[1];
if(thn >= 3) { printf("NO of theta: MORE THAN 2"); goto lab41; }
if(thbarn >= 3) { printf("NO of theta-bar: MORE THAN 2"); goto lab41; }
SORTOUTthBth();
thBthBthth();
SigmaContraction();
VecSufChange();
lab41:;
```

/* thn: No of θ within a term; thbarn: No of $\bar{\theta}$ within a term; */

10.2 termscombine

printf("*** TERMSCOMBINE ***");

[Initialization]
c=0,1,$\cdots$,maxcompno2-1
type1[c]='x'
th1[c][ ][ ]=99
si1[c][ ][ ]=99
dpsv1[c]=99
psi1[c][ ][ ]=99
dps1[c][ ][ ]=99
B1[c][ ]=99
A1[c][ ]=99
F1[c][ ]=99
C1[c][ ]=99

(T(0)term of sf=0 SuperField)$\times$
(T(1)term of sf=1 SuperField)$\times$
.
.
(T(sfieldno-1)term of sf=sfieldno-1 SuperField)

$\longrightarrow$ One Term Combined

Indices of a combined term
"Gathered Term Check"
No of Components, weight1[0], weight1[1], DIFnum1, THnum1[ ], SIGnum[ ], PSInum1[ ], Bnum[ ], Anum1[ ], Fnum1[ ], Cnum1[ ]

10.3 SORTOUTthBth

Moving th(θ), thbar($\bar{\theta}$), si(σ) and sibar($\bar{\sigma}$) *to the biggining part of a term.* In the process, we obtain the sign change due to the Grassman algebra.

PlusMinus : Sign change

Here we use the function: idno(TP).

idno(TP); a function which produces FieldID number correponding to a type TP.

10.4 thBthBthth

First we change the data form of th[ ][ ][ ]. From :

th[c][i][j]=α, i,j=0 or 1 c=0,1,$\cdots$,nth-1

to an another form:

I[c]=i, J[c]=j, Suf[c]=α

In this part, the program classifies terms by the power of θ and $\bar{\theta}$.
(1) indipendent of θ and $\bar{\theta}$ (nth=0)
(2) $\theta^{\alpha}, \theta_{\alpha}, \bar{\theta}_{\dot{\alpha}}, \bar{\theta}^{\dot{\alpha}}$ (nth=1)
(3) $\theta^{\alpha}\bar{\theta}^{\dot{\beta}}, \theta^{\alpha}\bar{\theta}_{\dot{\beta}}, \cdots$ (nth=2)
(4) $\theta^{\alpha}\theta^{\beta}, \theta^{\alpha}\theta_{\beta}, \cdots; \bar{\theta}^{\dot{\alpha}}\bar{\theta}^{\dot{\beta}}, \bar{\theta}^{\dot{\alpha}}\bar{\theta}_{\dot{\beta}}, \cdots; \theta^2; \bar{\theta}^2$ (nth=2)
(5) $\theta^2(\bar{\theta}_{\dot{\alpha}}, \bar{\theta}^{\dot{\alpha}}), \bar{\theta}^2(\theta_{\alpha}, \theta^{\alpha})$ (nth=3)
(6) $\theta^{\alpha}\theta^{\beta}\bar{\theta}_{\dot{\gamma}}, \cdots; \bar{\theta}^{\dot{\alpha}}\bar{\theta}^{\dot{\beta}}\theta_{\gamma}, \cdots$ (nth=3)
(7) $\theta^{\alpha}\bar{\theta}^{\dot{\beta}}\bar{\theta}^{\dot{\gamma}}\theta^{\delta}, \cdots; \bar{\theta}^{\dot{\alpha}}\theta^{\beta}\theta^{\gamma}\bar{\theta}^{\dot{\delta}}, \cdots; \cdots$ (nth=4)
(8) $\theta^{\alpha}\bar{\theta}^{\dot{\beta}}\theta^{\gamma}\bar{\theta}^{\dot{\gamma}}, \cdots; \bar{\theta}^{\dot{\alpha}}\theta^{\beta}\bar{\theta}^{\dot{\gamma}}\theta^{\delta}, \cdots; \cdots$ (nth=4)
(9) $\theta^{\alpha}\theta^{\beta}\bar{\theta}^{\dot{\gamma}}\bar{\theta}^{\dot{\delta}}, \cdots; \bar{\theta}^{\dot{\alpha}}\bar{\theta}^{\dot{\beta}}\theta^{\gamma}\theta^{\delta}, \cdots; \cdots$ (nth=4)

The formula presented in Sec.5 is exploited here. As the output, we obtain

[OutPut] weight, PlusMinus, Sign, Nthth, NthBthB, Nhalf

This OutPut means the following factor

$$(\text{weight}) * (-1)^{\text{PlusMinus+Sign}} * (\frac{1}{2})^{\text{Nhalf}} * (\theta^2)^{\text{Nthth}} (\bar{\theta}^2)^{\text{NthBthB}}$$

appears in the process of this subsection. 'weight' is the starting one: weight=weight[0]+i weight[1]. Here we use SufChange().

SufChange(): Chiral suffix contraction using δ^{β}_{α}, $\epsilon^{\alpha\beta}$, $\epsilon_{\alpha\beta}$. It is done for component c with c≥nth or c≥nth+nsi.

10.5 SigmaContraction

This routine is composed of 5 parts, (A),(B),(C),(D) and (E).

(A) Change of the data form
First we change the data form: (N=nsi-1)

$\sigma^{m_0}_{\alpha_0\dot{\alpha}_0}$ | $\sigma^{m_1\alpha_1\dot{\alpha}_1}$ | ... | $\sigma^{m_N\ \dot{\alpha}_N}_{\alpha_N}$

si[nth][2][2]	si[nth+1][2][2]	...	si[nth+nsi-1][2][2]
siv[nth]	siv[nth+1]		siv[nth+nsi-1]

to an another form:

c=0	c=1		c=N
Csuff[0][0]=α_0	Csuff[1][0]=α_1		Csuff[N][0]=α_N
Csuff[0][1]=$\dot{\alpha}_0$	Csuff[1][1]=$\dot{\alpha}_1$		Csuff[N][1]=$\dot{\alpha}_N$
Vsuff[0]=m_0	Vsuff[1]=m_1		Vsuff[N]=m_N
UD[0][0]=1(down)	UD[1][0]=0(up)		UD[N][0]=1(down)
UD[0][1]=1(down)	UD[1][1]=0(up)	...	UD[N][1]=0(up)

(B) Chiral-suffix-pair search

Search for same suffixes in {Csuff[c][0]|c=0,1,$\cdots$,N}(odd number suffixes) and in {Csuff[c][1]|c=0,1,$\cdots$,N}(even number suffixes). When Csuff[c1][0]=Csuff[c2][0], then we call (c1,c2) chiral-suffix-pair (or, simply, Cpair). For the odd number case, we list them as

CpairO[cpairnoO=0][0]=c1
CpairO[cpairnoO=0][1]=c2 (c1<c2)
CpairO[cpairnoO=0][2]=α_0(common chiral suffix)
CpairO[cpairnoO=1][0]=c3
CpairO[cpairnoO=1][1]=c4 (c3<c4)
CpairO[cpairnoO=1][2]=α_1
.
.
CpairO[NcpairO-1][0]=c*
CpairO[NcpairO-1][1]=c+ (c*<c+)
CpairO[NcpairO-1][2]=$\alpha_{NcpairO-1}$

For the even number case,

CpairE[cpairnoE=0][0]=c10
CpairE[cpairnoE=0][1]=c11 (c10<c11)
CpairE[cpairnoE=0][2]=$\dot{\alpha}_0$(common anti-chiral suffix)
CpairE[cpairnoE=1][0]=c12
CpairE[cpairnoE=1][1]=c13 (c12<c13)
CpairE[cpairnoE=1][2]=$\dot{\alpha}_1$
.
.
CpairE[NcpairE-1][0]=c**
CpairE[NcpairE-1][1]=c++ (c**<c++)
CpairE[NcpairE-1][2]=$\dot{\alpha}_{NcpairE-1}$

(C) Grouping σ's

A group of σ's is defined to be a set of σ's which are connected by chiral-suffix or anti-chiral-suffix contractions. The number of groups within a term is assigned to be **GrNum**. This an important Graph index. The number of σ's which each group has is stored in **SigmaN**[GR] (GR=0,1,$\cdots$,GrNum-1). Each group consits of some σ's specified their component numbers. They are stored in **Group**[GR][nseries] (nseries=0,1,$\cdot$,SigmaN[GR]-1). In this process of grouping, the program traces the CpairO and CpairE defined in (B). Each group is determined by a series of cpairnoO's and cpainoE's. They are stored in **CPseries**[GR][ ].

(D) Vector-suffix-pair search

Search the same vector-suffix in {Vsuff[c]|c=0,1,$\cdots$,N}.

```
Vpair[0][0]=c1, Vpair[0][1]=c2
Vpair[1][0]=c3, Vpair[1][1]=c4
     .          .
     .          .
Vpair[vpairno-1][0]=c*, Vpair[vpairno-1][1]=c+
```

(E) Classifying σ's part by indices (nsi, vpairno, NcpairO, NcpairE, closed chiral-loop)

Using graph indices defined in the Sec.4, we can classify σ's part. See the TABLE 2,3 and 4 in the text. Each class (we call here the finally classified place "class") has the number SigGraN which is the set of the graph indices. In each class, we reduce the expression using the graphical relations explained in Sec.7. Here the following important data are obtained and transfered to VecSufChange().

BranchN: Generally each class decomposes to some branches after the reduction using the graphical relations. We specify the number of branches.
NoChangeN[br]: For each branch(br), NoChangeN is assigned, 0 for the case that σ's part does not change, 1 for the case that σ's part changes.
RedSigN[br]: For each branch, the decrease-number of σ's after the reduction is stored. This number is important for adjusting the output form.
cstart: Normally this is fixed as cstart=nth+nsi
LorenContN[br]: For each branch, specify the number of Lorenz contraction, in other words, the number of η^{mn}.
SUFF[br][10][2]: Specify the vector suffixes appearing in η's.
MultiFac[br][2]: For each branch, a multiplicative factor is specified.
si1BR[br][c][2][2], siv1BR[br][c]: For each branch, specify the σ's part.

ep1[c][4]: For each branch, specify the totally-antisymmetric tensor when this term appears.

10.6 VecSufChange

Vector suffix contraction is peformed using $\eta^{mn}\eta^{ls}\cdots$. It is done for component c (c≥cstart=nth+nsi).

10.6.1 FinalOutPut

11 Appendix B: Input and Output Examples

11.1 Wess-Zumino Model

We demonstrate the calculation of $\Phi^\dagger\Phi$, where Φ is the chiral superfield, and obtain the component form. The next input data can be read from the graphical eqauations (6) and (7).

INPUT DATA

```
   2
   6
   0 -1  4  t  0 0 1  s  0 1 1  1 1 2  51  t  1 0 2  B  1 51
   1 0  5  t  0 0 1  t  0 1 1  t  1 1 2  t  1 0 2  C  1 1
   2 0  2  t  1 1 2  p  1 0 2
   0 1  5  t  1 1 2  t  1 0 2  t  0 0 1  s  0 1 1  1 1 4  51  q  1 0 4
51
   1 0  3  t  1 1 2  t  1 0 2  F  1 1
   1 0  1  A  1 1
   6
   0 1  4  t  0 0 5  s  0 1 5  1 1 6  52  t  1 0 6  B  0 52
   1 0  5  t  0 0 5  t  0 1 5  t  1 1 6  t  1 0 6  C  0 1
   2 0  2  t  0 0 5  p  0 1 5
   0 -1  5  t  0 0 5  t  0 1 5  q  0 0 7  52  s  0 1 7  1 1 6  52  t  1
0 6
   1 0  3  t  0 0 5  t  0 1 5  F  0 1
   1 0  1  A  0 1
```

OUT PUT

T[0]=0 T[1]=0

****** TERMSCOMBINE ****
th*th*thbar*thbar-term

****** SORTOUTthBth ****
lab50 at thBthBthth, Final result
weight= 1+i(0) PlusMinus= 0 Sign=2 Nthth=1 NthBthB=1 Nhalf=2

****** SigmaContraction ****
SigGraN=201199 FinalOutPut: MultiFac=-2 + i(0) type2[c=6]= B B2[c=6,1]=51
type2[c=7]= B B2[c=7,0]=51

T[0]=1 T[1]=5

****** TERMSCOMBINE ****
th*th*thbar*thbar-term

****** SORTOUTthBth ****
lab50 at thBthBthth, Final result
weight= 1+i(0) PlusMinus= 0 Sign=0 Nthth=1 NthBthB=1 Nhalf=0

****** SigmaContraction ****
SigGraN=0 FinalOutPut: MultiFac=1 + i(0) type2[c=4]= C C2[c=4,1]=1
type2[c=5]= A A2[c=5,0]=1

T[0]=2 T[1]=3

****** TERMSCOMBINE ****
th*th*thbar*thbar-term

****** SORTOUTthBth ****
lab50 at thBthBthth, Final result
weight= 0+i(-2) PlusMinus= 4 Sign=0 Nthth=1 NthBthB=1 Nhalf=1

****** SigmaContraction ****
SigGraN=1 FinalOutPut: MultiFac=1 + i(0) type2[c=4]= s si2[c=4,0,1]=7
si2[c=4,1,1]=2 siv2[c=4]=52 type2[c=5]= p psi2[c=5,1,0]=2 type2[c=6]=
q dps2[c=6,0,0]=7 dpsv2[c=6]=52

T[0]=3 T[1]=2

****** TERMSCOMBINE ****
th*th*thbar*thbar-term

****** SORTOUTthBth **** lab50 at thBthBthth, Final result
weight= 0+i(2) PlusMinus= 1 Sign=1 Nthth=1 NthBthB=1 Nhalf=1

****** SigmaContraction ****
SigGraN=1 FinalOutPut: MultiFac=1 + i(0) type2[c=4]= s si2[c=4,0,1]=1
si2[c=4,1,1]=4 siv2[c=4]=51 type2[c=5]= q dps2[c=5,1,0]=4 dpsv2[c=5]=51
type2[c=6]= p psi2[c=6,0,0]=1

T[0]=4 T[1]=4

****** TERMSCOMBINE ****
th*th*thbar*thbar-term

****** SORTOUTthBth ****
lab50 at thBthBthth, Final result
weight= 1+i(0) PlusMinus= 0 Sign=0 Nthth=1 NthBthB=1 Nhalf=0

****** SigmaContraction ****
SigGraN=0 FinalOutPut: MultiFac=1 + i(0) type2[c=4]= F F2[c=4,1]=1
type2[c=5]= F F2[c=5,0]=1

T[0]=5 T[1]=1

****** TERMSCOMBINE ****
th*th*thbar*thbar-term

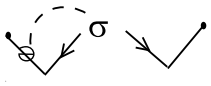
****** SORTOUTthBth ****
lab50 at thBthBthth, Final result
weight= 1+i(0) PlusMinus= 0 Sign=0 Nthth=1 NthBthB=1 Nhalf=0

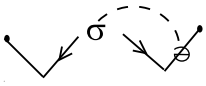
****** SigmaContraction ****
SigGraN=0 FinalOutPut: MultiFac=1 + i(0) type2[c=4]= A A2[c=4,1]=1
type2[c=5]= C C2[c=5,0]=1

Gathering all terms, we obtain

$$\Phi^\dagger\Phi|_{\theta^2\bar\theta^2} == -\frac{1}{2}\partial_m A^*\partial^m A\ (\mathbf{0},\mathbf{0}) + \frac{1}{4}\partial^2 A^*\cdot A\ (\mathbf{1},\mathbf{5}) + \frac{1}{4}A^*\partial^2 A\ (\mathbf{5},\mathbf{1})$$

$$-i\times(-1)\ \ (\mathbf{2},\mathbf{3}) + i\times(-1)\ \ (\mathbf{3},\mathbf{2}) + F^*F\ (\mathbf{4},\mathbf{4}) \quad (10)$$





This is the Wess-Zumino Lagrangian. We donot ignore the total divergence here.

11.2 Super QED

The calculation of $W_\alpha W^\alpha$ gives the kinetic terms of the photon and the photino in Super QED. The following input data can be read from the graphical eqauations (8) and (9).

INPUT DATA

```
2
4
0 -1  1  l  0 1 1
1 0  2  t  0 1 1  D  1
0 -1  4  s  0 1 1  1 1 2  51  s  0 0 3  1 0 2  52  t  0 1 3  w  52
51
1 0  4  t  0 0 3  t  0 1 3  s  0 1 1  1 1 2  51  m  1 0 2  51
4
0 -1  1  l  0 0 1
1 0  2  t  0 0 1  D  1
0 -1  4  s  0 0 1  1 1 4  53  s  0 0 5  1 0 4  54  t  0 1 5  w  54
53
1 0  4  t  0 0 5  t  0 1 5  s  0 0 1  1 1 4  53  m  1 0 4  53
```

OUT PUT

```
T[0]=0 T[1]=3

****** TERMSCOMBINE ****
th*th-term

****** SORTOUTthBth ****
lab50 at thBthBthth, Final result
```

weight= 0+i(-1) PlusMinus= 2 Sign=0 Nthth=1 NthBthB=0 Nhalf=0

****** SigmaContraction ****
SigGraN=1 FinalOutPut: MultiFac=1 + i(0) type2[c=2]= s si2[c=2,0,0]=1
si2[c=2,1,1]=4 siv2[c=2]=53 type2[c=3]= l la2[c=3,0,1]=1 type2[c=4]=
m dl2[c=4,1,0]=4 dlv2[c=4]=53

T[0]=1 T[1]=1

****** TERMSCOMBINE ****
th*th-term

****** SORTOUTthBth ****
lab50 at thBthBthth, Final result
weight= 1+i(0) PlusMinus= 0 Sign=1 Nthth=1 NthBthB=0 Nhalf=0

****** SigmaContraction ****
SigGraN=0 FinalOutPut: MultiFac=1 + i(0) type2[c=2]= D D2[c=2]=1
type2[c=3]= D D2[c=3]=1

T[0]=1 T[1]=2

****** TERMSCOMBINE ****
th*th-term

****** SORTOUTthBth ****
lab50 at thBthBthth, Final result
weight= 0+i(-1) PlusMinus= 0 Sign=0 Nthth=1 NthBthB=0 Nhalf=1

****** SigmaContraction ****
SigGraN=201199 FinalOutPut: MultiFac=2 + i(0) type2[c=4]= D D2[c=4]=1
type2[c=5]= w dv2[c=5]=53 dvv2[c=5]=53

T[0]=2 T[1]=1

****** TERMSCOMBINE ****
th*th-term

****** SORTOUTthBth ****
lab50 at thBthBthth, Final result
weight= 0+i(-1) PlusMinus= 0 Sign=1 Nthth=1 NthBthB=0 Nhalf=1

****** SigmaContraction ****
SigGraN=201199 FinalOutPut: MultiFac=-2 + i(0) type2[c=4]= w dv2[c=4]=51 dvv2[c=4]=51 type2[c=5]= D D2[c=5]=1

T[0]=2 T[1]=2

****** TERMSCOMBINE ****
th*th-term

****** SORTOUTthBth ****
lab50 at thBthBthth, Final result
weight= -1+i(0) PlusMinus= 0 Sign=0 Nthth=1 NthBthB=0 Nhalf=1

****** SigmaContraction ****
SigGraN=402299
Branch No 0
FinalOutPut: MultiFac=-2 + i(0) type2[c=6]= w dv2[c=6]=51 dvv2[c=6]=51 type2[c=7]= w dv2[c=7]=54 dvv2[c=7]=54
Branch No 1
FinalOutPut: MultiFac=2 + i(0) type2[c=6]= w dv2[c=6]=52 dvv2[c=6]=51 type2[c=7]= w dv2[c=7]=51 dvv2[c=7]=52
Branch No 2
FinalOutPut: MultiFac=-2 + i(0) type2[c=6]= w dv2[c=6]=52 dvv2[c=6]=51 type2[c=7]= w dv2[c=7]=52 dvv2[c=7]=51
Branch No 3
FinalOutPut: MultiFac=0 + i(2) type2[c=5]= e ep(51,52,54,53) type2[c=6]= w dv2[c=6]=52 dvv2[c=6]=51 type2[c=7]= w dv2[c=7]=54 dvv2[c=7]=53

T[0]=3 T[1]=0

****** TERMSCOMBINE ****
th*th-term

****** SORTOUTthBth ****
lab50 at thBthBthth, Final result
weight= 0+i(-1) PlusMinus= 0 Sign=0 Nthth=1 NthBthB=0 Nhalf=0

****** SigmaContraction ****
SigGraN=1 FinalOutPut: MultiFac=1 + i(0) type2[c=2]= s si2[c=2,0,1]=1 si2[c=2,1,1]=2 siv2[c=2]=51 type2[c=3]= m dl2[c=3,1,0]=2 dlv2[c=3]=51

type2[c=4]= l la2[c=4,0,0]=1

Gathering all terms, we obtain

$$W_\alpha W^\alpha = 2i \quad \{(\mathbf{0},\mathbf{3}),(\mathbf{3},\mathbf{0})\} \ + \ 0\ (\mathbf{1},\mathbf{2}) \ + \ 0\ (\mathbf{2},\mathbf{1})$$

$$-\frac{1}{2}\left(0\ (\mathbf{2},\mathbf{2})\mathbf{br0} - \frac{1}{2}{v_{mn}}^2\ (\mathbf{2},\mathbf{2})\mathbf{br1} - \frac{1}{2}{v_{mn}}^2\ (\mathbf{2},\mathbf{2})\mathbf{br2} - 2i\epsilon^{m,n,s,l}\frac{1}{2}v_{mn}\cdot\frac{1}{2}v_{sl}\ (\mathbf{2},\mathbf{2})\mathbf{br3}\right)$$
$$-D^2\ (\mathbf{1},\mathbf{1}). \quad (11)$$

Adding $\bar{W}\bar{W}$, we obtain the Lagrangian as

$$\mathcal{L} = \frac{1}{4}(-W_\alpha W^\alpha|_{\theta^2} + \bar{W}_{\dot\alpha}\bar{W}^{\dot\alpha}|_{\bar\theta^2})$$

$$= -\frac{1}{4}{v_{mn}}^2 + \frac{i}{2}(\quad - \quad) + \frac{1}{2}D^2\ , \quad (12)$$

where we do not ignore the total divergence. This is the kinetic term of the photon and the photino.

Acknowledgements

This work is completed in the author's stay at ITP, Univ. Wien. He thanks A. Bartl for reading carefully the manuscript and all menbers of the institute for the hospitality.

References

[1] S. Ichinose, hep-th/0301166, DAMTP-2003-8, US-03-01, "Graphical Representation of Supersymmetry"

[2] S. Ichinose, hep-th/0410027, Proc. 12th Int.Conf. on "Supersymmetry and Unification of Fundamental Interactions"(June 17-23,2004,Epochal Tsukuba Congress Center,Japan), p853-856, "Graphical Representation of Supersymmetry and Computer Calculation"

[3] S. Ichinose, Univ. Vienna preprint UWThPh-2006-7, "Graphical Representation of Supersymmetry"

[4] J. Wess and J. Bagger, *Supersymmetry and Supergravity*. Princeton University Press, Princeton, 1992

[5] G. 'tHooft and M. Veltman, Nucl.Phys.B44,189(1972)

[6] M.H. Goroff and A. Sagnotti, Phys.Lett.B150(1985)81; Nucl.Phys.B266(1986)709

[7] A.E.M. van de Ven, Nucl.Phys.B378(1992)309

[8] S. Ichinose, Int.Jour.Mod.Phys.**C9**(1998)243, hep-th/9609014